

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of

Inventors: Shuji KONDO  
Application No.: New Patent Application  
Filed: March 19, 2004  
For: MICROCONTROLLER

CLAIM FOR PRIORITY

Honorable Commissioner of  
Patents and Trademarks  
Washington, D.C. 20231

Sir:

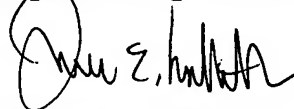
The benefit of the filing date of the following prior foreign application filed in the following foreign country is hereby requested for the above-identified application and the priority provided in 35 USC 119 is hereby claimed:

Japanese Appln. No. 2003-094686, filed March 31, 2003.

In support of this claim, a certified copy of said original foreign application is filed herewith.

It is requested that the file of this application be marked to indicate that the requirements of 35 USC 119 have been fulfilled and that the Patent and Trademark Office kindly acknowledge receipt of this document.

Respectfully submitted,



James E. Ledbetter  
Registration No. 28,732

Date: March 19, 2004

JEL/spp  
Attorney Docket No. L8462.04110  
STEVENS, DAVIS, MILLER & MOSHER, L.L.P.  
1615 L Street, NW, Suite 850  
P.O. Box 34387  
Washington, DC 20043-4387  
Telephone: (202) 785-0100  
Facsimile: (202) 408-5200

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年 3月31日  
Date of Application:

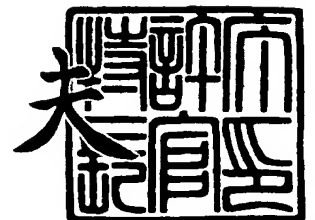
出願番号 特願2003-094686  
Application Number:  
[ST. 10/C]: [JP2003-094686]

出願人 松下電器産業株式会社  
Applicant(s):

2004年 2月10日

特許庁長官  
Commissioner,  
Japan Patent Office

今井 康夫



出証番号 出証特2004-3008283

【書類名】 特許願

【整理番号】 5038340134

【あて先】 特許庁長官殿

【国際特許分類】 G06F 1/08

【発明者】

    【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地  
                        松下電器産業株式会社内

    【氏名】 近藤 秀二

【特許出願人】

    【識別番号】 000005821

    【氏名又は名称】 松下電器産業株式会社

【代理人】

    【識別番号】 100076174

    【弁理士】

    【氏名又は名称】 宮井 暎夫

【選任した代理人】

    【識別番号】 100105979

    【弁理士】

    【氏名又は名称】 伊藤 誠

【手数料の表示】

    【予納台帳番号】 010814

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

    【包括委任状番号】 0212624

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マイクロコントローラ

【特許請求の範囲】

【請求項 1】 発振周波数の  $1/n$  ( $n$  は分周比で 1 以上の整数) の周波数であって異なる周波数の複数のクロック信号の中から周波数情報に応じた前記クロック信号をシステムクロックとして入力し、前記システムクロックに同期して動作し、メモリに格納されたオペレーションコードとオペランドからなる命令プログラムに対して、パイプライン処理により前記オペレーションコードと前記オペランドをフェッチして解読し、その解読結果に基づいて前記命令プログラムを実行するプロセッサを備え、

前記オペレーションコードは前記システムクロックの分周比を決定する周波数制御信号が付加されて前記メモリに格納されており、

前記プロセッサは、前記周波数制御信号を前記オペレーションコードとともにパイプライン処理によりフェッチ、解読することにより前記周波数制御信号に応じた前記周波数情報を出力することを特徴とするマイクロコントローラ。

【請求項 2】 発振周波数の  $1/n$  ( $n$  は分周比で 1 以上の整数) の周波数であって異なる周波数の複数のクロック信号の中から周波数情報に応じた前記クロック信号をシステムクロックとして入力し、前記システムクロックに同期して動作し、メモリに格納された拡張コードとオペレーションコードとオペランドからなる命令プログラムに対して、パイプライン処理により前記拡張コードと前記オペレーションコードと前記オペランドをフェッチして解読し、その解読結果に基づいて前記命令プログラムを実行するプロセッサを備え、

マシンコードによって前記オペレーションコードを各分周比に対応してページに分類された複数ページからなる命令マップとして構成し、前記オペレーションコードとともにパイプライン処理される前記拡張コードが前記命令マップのページを示し、

前記プロセッサは、前記拡張コードをパイプライン処理によりフェッチ、解読することにより前記拡張コードが示す前記命令マップのページに対応する分周比に応じた前記周波数情報を出力することを特徴とするマイクロコントローラ。

【請求項 3】 前記発振周波数の  $1/n$  ( $n$  は分周比で 1 以上の整数) の周波数であって異なる周波数の複数のクロック信号を生成し、その中から前記プロセッサから出力される前記周波数情報に応じたクロック信号を選択して前記プロセッサへ前記システムクロックとして出力するクロックジェネレータを設けた請求項 1 または 2 記載のマイクロコントローラ。

【請求項 4】 前記クロックジェネレータは、  
前記発振周波数の  $1/n$  ( $n$  は分周比で 1 以上の整数) の周波数であって異なる周波数の複数のクロック信号を出力する分周手段と、  
前記分周手段の出力する前記複数のクロック信号の中から選択信号に基づいて 1 つのクロック信号を選択し前記システムクロックとして出力する選択手段と、  
前記プロセッサから出力される前記周波数情報に対応する前記選択信号を前記選択手段に出力する制御手段とを有した請求項 3 記載のマイクロコントローラ。

【請求項 5】 ソースプログラムに記述された前記システムクロックの分周比を決定する分周比設定記述に従って前記周波数制御信号を生成し、前記分周比設定記述の後段の命令から生成される前記オペレーションコードに付加し ROM コード化する手段を設けた請求項 1 記載のマイクロコントローラ。

【請求項 6】 ソースプログラムに記述された前記システムクロックの分周比を決定する前記分周比設定記述に従って前記周波数制御信号を生成し、前記分周比設定記述の前段の命令の実行サイクル数が基準未満であれば前記周波数制御信号を前記分周比設定記述の前段の命令から生成される前記オペレーションコードに付加し、基準以上であれば前記周波数制御信号を前記分周比設定記述の後段の命令から生成される前記オペレーションコードに付加し ROM コード化する手段を設けた請求項 1 記載のマイクロコントローラ。

【請求項 7】 ソースプログラムに記述された前記システムクロックの分周比を決定する分周比設定記述に従って分周比に対応する前記拡張コードを選択し、前記拡張コードが示す命令マップのページから前記分周比設定記述の後段の命令に相当する前記オペレーションコードを選択し ROM コード化する手段を設けた請求項 2 記載のマイクロコントローラ。

【発明の詳細な説明】

## 【0001】

## 【発明の属する技術分野】

本発明は、システムクロックに同期して動作し、ROM等のメモリ上の命令プログラムに対し、パイプライン処理によりフェッチ、解読および実行するマイクロコントローラに関する。

## 【0002】

## 【従来の技術】

低消費電力化の観点でマイクロコントローラを用いたシステムを見直した場合、すべてのタスクにおいてマイクロコントローラの最高周波数での動作が必要とされるわけではない。要は、システムのパフォーマンスを落とさない最低限の動作周波数であればよい。

## 【0003】

一般的なマイクロコントローラにおいてはこのような場合を想定し、原発振からシステムクロックを生成する過程で分周比を変更できる構成としている。

## 【0004】

従来のマイクロコントローラでは、分周比の制御はアドレスマッピングされたレジスタへの任意データの設定によって行われることが多いが、このような構成の場合、消費電力、動作周波数を細やかな制御を試みても書き込みのための命令実行サイクル分の時間的ロスが発生してしまうことが障害になり、意図した周波数制御および電力制御の効果が得られない場合がある。これら課題の対策として、一方ではアクセスしたメモリ空間に応じて周波数を変更しているものもある（例えば特許文献1参照）。しかしながら、変更すべき周波数に割り当てられたメモリ空間へのジャンプ命令の実行が必要になるためいずれにしても完全な課題解決には至っていない。

## 【0005】

以上のような従来のマイクロコントローラについて、以下に説明する。図10、図11は、従来のマイクロコントローラの概略構成を示すブロック図であり、図10はその内部のクロックジェネレータの構成を詳細に示し、図11はその内部のCPUの構成を詳細に示したものである。

## 【0006】

図10において、1000はマイクロコントローラであり、100はCPUである。ROM700とはバスを介して接続され、クロックジェネレータ800とはバスを介した接続以外にクロックジェネレータ800からシステムクロック `sysclk` が入力されマイクロコントローラ1000はこの `sysclk` に同期して動作する。クロックジェネレータ800は原発振 `oscin` を入力とし、クロック分周回路801、セクタ802、クロック分周制御回路803、クロック分周制御レジスタ804から構成される。クロック分周回路801は `oscin` を分周し、1分周信号の他複数の分周信号を生成する。クロック分周制御レジスタ804はCPU100の処理する命令によってリード／ライト可能なレジスタで設定されるデータによってクロック分周回路801で生成される分周信号のうちひとつを選択する情報を保持する。クロック分周制御回路803はクロック分周制御レジスタ804に設定されたデータをもとにクロック切り替えのタイミングを調整し、セクタ802の制御信号 `oscsel` を出力する。セクタ802は `oscsel` に従いクロック分周回路801から出力される分周信号の中からひとつを選択し、`sysclk` としてCPU100に伝達する。

## 【0007】

図11において、CPU100は、命令デコーダ400、データバス300、データレジスタ500、アドレスレジスタ600、バスインターフェイス200から構成され、CPU内部でのデータのやり取りはバスインターフェイス200を介して行い、CPUの動作はマイクロコード(MIR)によって制御される。次に、上記のように構成されたマイクロコントローラについて、8ビットマイクロコントローラを例にその一般的な動作を以下に説明する。

## 【0008】

まず、ROM700から入力されたデータは、バスインターフェイス200を経由して、命令デコーダ400内のインストラクションフェッチバッファ(IFB)401に取り込まれ、インストラクションバッファ(IB)402を経由し、その後、オペレーションコードとオペランドに分割される。

## 【0009】

IB402から出力されたオペレーションコードは、インストラクションレジスタ（IR）403に入力された後、プログラマブルロジックアレイ（PLA）404に解読され、MIRとしてバスインターフェイス200、データパス300、命令デコーダ400の各構成ブロックへ供給され、各構成ブロックは、入力されたMIRに従った処理動作を実行する。

#### 【0010】

またIB402から出力されたオペランドは、MIRに従って、データパス300あるいはデータレジスタ600やアドレスレジスタ700などへ伝達される。

#### 【0011】

システムクロック `sysclk` の分周比を変更するにはアドレスマッピングされたクロック分周制御レジスタ804に対して各分周比に相当する設定値を書き込むことでクロック分周制御回路803で切り替えタイミングを調整し、選択信号 `osc sel` をセレクタ802に出力し、それに応じてセレクタ802では分周信号を選択し、`sysclk` としてCPU100に伝達する。

#### 【0012】

以上のような従来のマイクロコントローラの動作をプログラム例、タイミングチャートを用いて説明する。

#### 【0013】

図12の（a）、（b）、（c）は、従来のマイクロコントローラの命令フォーマット、プログラム例、動作タイミングチャートである。

#### 【0014】

命令フォーマットは、命令マップのページ数を示す4ビットの拡張コードと、8ビットのオペコード（オペレーションコード）と、 $4 \times n$ ビット（ $n=1, 2, \dots$ ）のオペランドとで構成される。

#### 【0015】

プログラム例は①から⑦の命令を実行するものとし、`sysclk` を `osc i` n の1分周で命令①を実行した後、命令②により分周比を1分周から2分周に切り替える。命令③および④を実行した後、命令⑤により再度1分周に戻し命令⑥



および命令⑦を実行する。タイミングチャート中に示される記号で①-1はプログラム例の命令①の1ニブル目のマシンコードを示し、① $\mu$ -1は命令①の実行サイクル1サイクル目を表すものとする。T1、T2は各々sysclkの立下りエッジ、立ち上がりエッジのタイミングを示す。タイミングAにおいてoscselが‘L’であるためsysclkは1/10scinの周期で動作する。IFB401、IB402はT1で命令①-1と①-2をROM700からフェッチする。命令①-1は拡張コード、①-2はオペコードの1ニブル目でありオペランドではないためいずれのデータもIR403に対して出力する。タイミングBにおいてIR403はT2で①-1, 2をラッチし、PLA404に出力しPLA404はデコードを開始する。①-1は拡張コードであり、①-2はオペコードの2ニブル目がフェッチされるまで処理されないため、タイミングCにおいて次のT1では①-1に相当するMIR (① $\mu$ -1) のみを拡張コードの認識サイクルとして出力する。また同時にIFB401は命令①-3と②-1をフェッチする。タイミングDにおいてIR403は命令①の残り1ニブルのオペコード①-3を取り込み、処理待ちの①-2とともにPLA404にてデコードする。タイミングEにおいては①-2, 3のMIR (① $\mu$ -2) を出力するとともに②-2, 3をIFB401がフェッチする。また、タイミングDでIR403に取り込まれなかった②-1はタイミングEでIB402にシフトする。なお、③-1のように拡張コードがIFB401、IB402を経由した場合にはその間に認識されるためタイミングFではIR403には取り込まれず、オペコードである③-2, 3がIR403に取り込まれる。

#### 【0016】

命令②では実行サイクル② $\mu$ -3でのクロック分周制御レジスタ804への書き込みによってタイミングGでoscselが‘H’になり、sysclkの周期を1/10scinから1/20scinに変更する。同様に命令⑤によってsysclkの周期を1/20scinから1/10scinに戻す。

#### 【0017】

以上のように従来のマイクロコントローラはクロック分周制御レジスタ804への書き込み命令によって周波数を制御することができる。

【0018】

【特許文献1】

特開平02-118811号公報

【0019】

【発明が解決しようとする課題】

しかしながら上記のような従来のマイクロコントローラにおいては、周波数の変更のためにレジスタへの書き込み命令実行のために数サイクル分が必要となる。これによって、周波数制御による消費電力の調整をシステムのパフォーマンスを維持したままきめ細かく行うことができないという問題点を有していた。

【0020】

本発明は、上記従来の問題点を解決するもので、周波数変更に要するタイムラグを解消することができ、タスクに応じた周波数制御を的確かつ迅速に実行することにより、消費電力を低減化することができるマイクロコントローラを提供する。

【0021】

【課題を解決するための手段】

本発明の請求項1記載のマイクロコントローラは、発振周波数の $1/n$ （ $n$ は分周比で1以上の整数）の周波数であって異なる周波数の複数のクロック信号の中から周波数情報に応じたクロック信号をシステムクロックとして入力し、システムクロックに同期して動作し、メモリに格納されたオペレーションコードとオペランドからなる命令プログラムに対して、パイプライン処理によりオペレーションコードとオペランドをフェッチして解読し、その解読結果に基づいて命令プログラムを実行するプロセッサを備え、オペレーションコードはシステムクロックの分周比を決定する周波数制御信号が付加されてメモリに格納されており、プロセッサは、周波数制御信号をオペレーションコードとともにパイプライン処理によりフェッチ、解読することにより周波数制御信号に応じた周波数情報を出力することを特徴とする。

【0022】

この構成によれば、周波数を制御する情報はパイプラインの中でオペレーショ

ンコードと同様に処理されるため、従来のように周波数制御レジスタへ設定データを書き込む必要がなく、システムのパフォーマンスを落とさず周波数制御をきめ細かく行い、マイクロコントローラの低消費電力化を効果的に実現することができる。

#### 【0023】

また、本発明の請求項2記載のマイクロコントローラは、発振周波数の $1/n$ （ $n$ は分周比で1以上の整数）の周波数であって異なる周波数の複数のクロック信号の中から周波数情報に応じたクロック信号をシステムクロックとして入力し、システムクロックに同期して動作し、メモリに格納された拡張コードとオペレーションコードとオペランドからなる命令プログラムに対して、パイプライン処理により拡張コードとオペレーションコードとオペランドをフェッチして解読し、その解読結果に基づいて命令プログラムを実行するプロセッサを備え、マシンコードによってオペレーションコードを各分周比に対応してページに分類された複数ページからなる命令マップとして構成し、オペレーションコードとともにパイプライン処理される拡張コードが命令マップのページを示し、プロセッサは、拡張コードをパイプライン処理によりフェッチ、解読することにより拡張コードが示す命令マップのページに対応する分周比に応じた周波数情報を出力することを特徴とする。

#### 【0024】

この構成によれば、既存の拡張コードエリアが周波数制御情報をもつため、回路規模の増加はなく、オペレーションコードより先にデコードされることにより周波数変更すべき命令の実行サイクルと周波数変更タイミングを合わせやすく、よりの確にプログラム開発者の意図に応じた低消費電力化が可能となる。

#### 【0025】

また、請求項3記載のマイクロコントローラのように、請求項1または2記載のマイクロコントローラにおいて、発振周波数の $1/n$ （ $n$ は分周比で1以上の整数）の周波数であって異なる周波数の複数のクロック信号を生成し、その中からプロセッサから出力される周波数情報に応じたクロック信号を選択してプロセッサへシステムクロックとして出力するクロックジェネレータを設けてあっても

よい。

#### 【0026】

また、このクロックジェネレータは、請求項4のように、発振周波数の $1/n$  ( $n$ は分周比で1以上の整数)の周波数であって異なる周波数の複数のクロック信号を出力する分周手段と、分周手段の出力する複数のクロック信号の中から選択信号に基づいて1つのクロック信号を選択しシステムクロックとして出力する選択手段と、プロセッサから出力される周波数情報に対応する選択信号を選択手段に出力する制御手段とを有した構成とすることができる。

#### 【0027】

また、請求項5記載のマイクロコントローラは、請求項1記載のマイクロコントローラにおいて、ソースプログラムに記述されたシステムクロックの分周比を決定する分周比設定記述に従って周波数制御信号を生成し、分周比設定記述の後段の命令から生成されるオペレーションコードに付加しROMコード化する手段を設けている。

#### 【0028】

これにより、周波数制御信号のオペレーションコードへの付加はプログラム開発者の負担なくコンパイラによって自動でマシンコード化が可能となる。

#### 【0029】

また、請求項6記載のマイクロコントローラは、請求項1記載のマイクロコントローラにおいて、ソースプログラムに記述されたシステムクロックの分周比を決定する分周比設定記述に従って周波数制御信号を生成し、分周比設定記述の前段の命令の実行サイクル数が基準未満であれば周波数制御信号を分周比設定記述の前段の命令から生成されるオペレーションコードに付加し、基準以上であれば周波数制御信号を分周比設定記述の後段の命令から生成されるオペレーションコードに付加しROMコード化する手段を設けている。

#### 【0030】

これにより、周波数制御信号がオペレーションコードとともにデコードされ実行されるまでの処理時間による周波数変更すべき命令の実行サイクルと周波数変更タイミングの誤差を削減することが可能となる。

## 【0031】

また、請求項7記載のマイクロコントローラは、請求項2記載のマイクロコントローラにおいて、ソースプログラムに記述されたシステムクロックの分周比を決定する分周比設定記述に従って分周比に対応する拡張コードを選択し、拡張コードが示す命令マップのページから分周比設定記述の後段の命令に相当するオペレーションコードを選択しROMコード化する手段を設けている。

## 【0032】

これにより、システムクロックの分周比の設定、その分周比に相当する命令コードの選択は固有の分周比設定記述によりプログラム開発者の負担なくコンパイラによって自動でマシンコード化が可能となる。

## 【0033】

## 【発明の実施の形態】

以下、本発明の実施の形態について、図面を参照しながら具体的に説明する。

## 【0034】

## (実施の形態1)

本発明の実施の形態1のマイクロコントローラについて説明する。図1、図2は、本実施の形態1のマイクロコントローラの概略構成を示すブロック図であり、図1はその内部のクロックジェネレータの構成を詳細に示し、図2はその内部のCPUの構成を詳細に示したものである。

## 【0035】

図1において、1000はマイクロコントローラであり、100はCPUである。CPU100とROM700とはバスを介して接続され、クロックジェネレータ800は原発振oscinを入力とし、クロック分周回路801、セクタ802、クロック分周制御回路803から構成され、CPU100から出力されるマイクロコードMIRからシステムクロックsysclkの分周比を決定し、CPU100に伝達する。マイクロコントローラ1000はこのsysclkに同期して動作する。クロック分周回路801は原発振oscinを分周し、1分周信号の他複数の分周信号を生成する。クロック分周制御回路803はCPU100から出力されたMIRをもとにクロック切り替えのタイミングを調整し、セ

レクタ 802 の制御信号 `osc sel` を出力する。セクタ 802 は `osc sel` に従いクロック分周回路 801 から出力される分周信号の中からひとつを選択し、`sys clk` として CPU100 に伝達する。

#### 【0036】

図 2 において、CPU100 は、命令デコーダ 400、データパス 300、データレジスタ 500、アドレスレジスタ 600、バスインターフェイス 200 から構成され、CPU 内部でのデータのやり取りはバスインターフェイス 200 を介して行い、CPU の動作はマイクロコード (MIR) によって制御される。

#### 【0037】

次に、上記のように構成された実施の形態 1 のマイクロコントローラの動作を説明する。

#### 【0038】

まず、ROM700 から入力されたデータは、バスインターフェイス 200 を介して命令デコーダ 400 内のインストラクションフェッチバッファ (IFB) 401 に取り込まれ、インストラクションバッファ (IB) 402 を経由し、その後オペレーションコードとオペランドに分割される。

#### 【0039】

IB402 から出力されたオペレーションコードは、インストラクションレジスタ (IR) 403 に入力された後、プログラマブルロジックアレイ (PLA) 404 に解読され、MIR としてバスインターフェイス 200、データパス 300、命令デコーダ 400 の各構成ブロックへ供給され、各構成ブロックは、入力された MIR に従った処理動作を実行する。

#### 【0040】

また、IB402 から出力されたオペランドは、MIR に従って、データパス 300 あるいはデータレジスタ 600 やアドレスレジスタ 700 などへ伝達される。

#### 【0041】

システムクロック `sys clk` の分周比を変更するメカニズムは、オペレーションコードに付加された周波数制御信号をオペレーションコードとともに解読し

、MIRとしてクロック分周制御回路803に伝達される。クロック分周制御回路803においては伝達されたMIRをもとに切り替えタイミングを調整し、選択信号`osc sel`をセレクタ802に出力し、セレクタ802で分周信号を選択し、`sys clk`としてCPU100に伝達する。

#### 【0042】

以上のような本実施の形態1のマイクロコントローラの動作をプログラム例、タイミングチャートを用いて説明する。

#### 【0043】

図3の(a)、(b)、(c)は、本実施の形態1のマイクロコントローラの命令フォーマット、プログラム例、動作タイミングチャートである。ここでは8ビットマイクロコントローラを例に`sys clk`の分周比を`osc in`の1分周から2分周そして再度1分周に切り替える動作を示している。

#### 【0044】

命令フォーマットは、命令マップのページ数を示す4ビットの拡張コードと、8ビットのオペコード（オペレーションコード）と、オペコードに付加した1ビットの周波数制御信号と、 $4 \times n$ ビット（ $n=1, 2 \dots$ ）のオペランドとで構成される。

#### 【0045】

プログラム例は①から⑦の命令を実行するものとし、命令③から`sys clk`の分周比を`osc in`の1分周から2分周に切り替え、命令⑥以降再度1分周に切り替える。オペコードに付加された周波数制御信号は\*0または\*1で表し、\*0のとき1分周を、\*1のとき2分周を選択するものとする。タイミングチャート中に示される記号で①-1はプログラム例の命令①の1ニブル目のマシンコードを示し、① $\mu$ -1は命令①の実行サイクルの1サイクル目を表すものとする。T1、T2は各々`sys clk`の立下りエッジ、立ち上がりエッジのタイミングを示す。

#### 【0046】

タイミングAにおいて`osc sel`が‘L’であるため`sys clk`は $1/1$  `osc in`の周期で動作する。IFB401、IB402はT1で命令①-1と

①-2をROM700からフェッチする。命令①-1は拡張コード、①-2はオペコードの1ニブル目でありオペランドではないためいずれのデータもIR403に対して出力する。タイミングBにおいてIR403はT2で①-1, 2をラッチし、PLA404に出力しPLA404はデコードを開始する。①-1は拡張コードであり、①-2はオペコードの2ニブル目がフェッチされるまで処理されないため、タイミングCにおいて次のT1では①-1に相当するMIR (① $\mu$ -1)のみを拡張コードの認識サイクルとして出力する。また同時にIFB401は命令①-3と③-1をフェッチする。タイミングDにおいてIR403は命令①の残り1ニブルのオペコード①-3を取り込み、処理待ちの①-2とともにPLA404にてデコードする。タイミングEにおいては①-2, 3のMIR (① $\mu$ -2)を出力するとともに③-2, 3をIFB401がフェッチする。また、タイミングDでIR403に取り込まれなかった③-1はタイミングEでIB402にシフトする。なお、③-1のように拡張コードがIFB401、IB402を経由した場合にはその間に認識されるためタイミングFではIR403には取り込まれず、オペコードである③-2, 3がIR403に取り込まれる。

#### 【0047】

③-3は付加された周波数制御信号に2分周制御の情報(\*1)を保有しているため③ $\mu$ -1のサイクル中にMIRとしてクロックジェネレータ800に出力される。クロック分周制御回路803は切り替えタイミングを調整しタイミングGでosc selを'H'としsys clkを1/2 osc inに切り替える。同様に命令⑥に付加された周波数制御信号によって1/1 osc inに戻す。

#### 【0048】

なお、付加される周波数制御信号は1ビットに限らずマイクロコントローラの仕様によって数ビットになる場合もある。

#### 【0049】

以上のように本実施の形態1によれば、オペレーションコードにシステムクロックの分周比を決定する周波数制御信号を付加することにより、オペレーションコードと同様にパイプラインの中で処理されるため、従来のように周波数制御レジスタ(804)へ設定データを書き込む時間を必要とせず、システムのパフォ



パフォーマンスを落とさずマイクロコントローラの低消費電力化を効果的に実現することができる。また、本実施の形態では8ビットマイクロコントローラを例に説明しているが、ビット数が多くなれば付加するビットの回路規模への影響は極めて小さなものになり、本発明の効果はさらに大きくなる。

#### 【0050】

(実施の形態2)

本発明の実施の形態2のマイクロコントローラについて説明する。

#### 【0051】

図4は、本実施の形態2のマイクロコントローラの概略構成を示すブロック図である。実施の形態1との違いはIFB401、IB402、IR403、PLA404と伝達されるオペレーションコードに周波数制御信号を付加しないため8ビットで構成される点のみで、その他は実施の形態1のマイクロコントローラと同様である。したがって、クロックジェネレータ800の内部構成は図1と同じである。

#### 【0052】

次に、上記のように構成された実施の形態2のマイクロコントローラの動作を説明する。

#### 【0053】

図5の(a)、(b)は、本実施の形態2のマイクロコントローラの命令マップ例である。命令マップは複数ページで構成され、分周比ごとに各ページに分類されている。したがって、命令マップのページ数を示す拡張コードによって同じオペレーションコードであっても異なる周波数で実行されることになる。例えば命令マップAと命令マップBはそれぞれ1分周と2分周に対応しており拡張コード‘0011’および‘0101’で分類している。マップ中のMC1とMC2はともに同じ実行内容の命令で拡張コード以外のマシンコードも同一である。これらのコードをCPUで処理した場合、同じ動作をMC1は1分周で行い、MC2は2分周で行うことになる。このように分周比に対応した拡張コードはパイプライン処理の中でオペコードと同様に解釈され、MIRとしてクロック分周制御回路803に伝達される。クロック分周制御回路803においては伝達されたM

IRをもとに切り替えタイミングを調整し、選択信号 `osc sel` をセクタ 802 に出力し、それに応じてセクタ 802 では分周信号を選択し、`sys clk` として CPU100 に伝達する。

#### 【0054】

以上のような本実施の形態2のマイクロコントローラの動作をプログラム例、タイミングチャートを用いて説明する。

#### 【0055】

図6の(a)、(b)、(c)は、本実施の形態2のマイクロコントローラの命令フォーマット、プログラム例、動作タイミングチャートである。ここでは実施の形態1のマイクロコントローラにおける動作説明と同様に8ビットマイクロコントローラを例に `sys clk` の分周比を `osc in` の1分周から2分周そして再度1分周に切り替える動作を示している。

#### 【0056】

命令フォーマットは、命令マップのページ数を示す4ビットの拡張コードと、8ビットのオペコード（オペレーションコード）と、 $4 \times n$  ビット（ $n=1, 2, \dots$ ）のオペランドとで構成される。実施の形態1のマイクロコントローラとの違いは周波数制御信号を付加していない点のみである。

#### 【0057】

プログラム例は①から⑦の命令を実行するものとし、命令③から `sys clk` の分周比を `osc in` の1分周から2分周に切り替え、命令⑥以降再度1分周に切り替える。拡張コードの‘0011’は `sys clk` を `osc in` の1分周に設定する命令であることを示し、‘0101’2分周であることを示す。

#### 【0058】

タイミングAにおいて `osc sel` が‘L’であるため `sys clk` は  $1/1$  `osc in` の周期で動作する。IFB401、IB402はT1で命令①-1と①-2をROM700からフェッチする。命令①-1は拡張コード、①-2はオペコードの1ニブル目でありオペランドではないためいずれのデータもIR403に対して出力する。タイミングBにおいてIR403はT2で①-1, 2をラッチし、PLA404に出力しPLA404はデコードを開始する。①-1は拡張

張コードであり、①-2はオペコードの2ニブル目がフェッチされるまで処理されないため、タイミングCにおいて次のT1では①-1に相当するMIR (① $\mu$ -1)のみを拡張コードの認識サイクルとして出力する。また同時にIFB401は命令①-3と③-1をフェッチする。タイミングDにおいてIR403は命令①の残り1ニブルのオペコード①-3を取り込み、処理待ちの①-2とともにPLA404にてデコードする。タイミングEにおいては①-2, 3のMIR (① $\mu$ -2)を出力するとともに③-2, 3をIFB401がフェッチする。また、タイミングDでIR403に取り込まれなかった③-1はタイミングEでIB402にシフトする。なお、③-1のように拡張コードがIFB401、IB402を経由した場合にはその間に認識されるためタイミングFではIR403には取り込まれず、オペコードである③-2, 3がIR403に取り込まれる。

#### 【0059】

③-1は拡張コードであり2分周制御の情報‘0101’を保有しており、IFB401、IB402を経由する間に解読されMIRとしてクロックジェネレータ800に出力される。クロック分周制御回路803は切り替えタイミングを調整しタイミングGでosc selを‘H’としsys clkを1/2 osc inに切り替える。同様に命令⑥の拡張コードによって1/1 osc inに戻す。拡張コードはオペコードより先にフェッチ、デコードされるため本来2分周で動作させたい③ $\mu$ -1、③ $\mu$ -2、④ $\mu$ -1、④ $\mu$ -2、1分周で動作させたい⑥ $\mu$ -1、⑦ $\mu$ -1、⑦ $\mu$ -2が想定どおりに実行することが可能である。

#### 【0060】

以上のように本実施の形態2によれば、拡張コードにシステムクロックの分周比を決定する情報を持たせることにより、オペレーションコードより早いサイクルでデコードすることが可能となるため、周波数を変更するためのタイムラグを最小に抑えることが可能となり、プログラム開発者は周波数制御をより簡単に正確に実行することが可能となる。したがって、周波数変更に要するタイムラグを解消しタスクに応じた周波数制御を的確かつ迅速に実行することによりマイクロコントローラの消費電力を低減することができる。

#### 【0061】

なお、命令は1分周と2分周の2種類に限らずマイクロコントローラの仕様によって数種類になる場合もある。また、命令数は分周比ごとに同数存在する必要はなく、例えばデフォルトの分周比に相当する命令に対してその他の分周比に相当する命令数は数個に限定することも可能である。

#### 【0062】

上記の実施の形態1と実施の形態2におけるマイクロコントローラ1000では、一般的なほとんどのマイクロコントローラと同様に、システムクロックを生成するクロックジェネレータ800を内蔵している構成としたが、クロックジェネレータ800は外付けであっても構わない。

#### 【0063】

(実施の形態3)

本発明の実施の形態3のマイクロコントローラについて説明する。

#### 【0064】

図7(a)は本実施の形態3のマイクロコントローラにおけるマシンコードの生成およびROMへの配置手順を示すフローチャート、図7(b)は本実施の形態3においてプログラムソースからコンパイラによってマシンコードを生成する方法を示す図である。図7(a)に示すように、プログラム開発によって作成されたプログラムソースは、通常コンパイラ等により発生されたデータファイルをもとに、ROMコードに変換された後、レイアウトパターンとして生成されROM配置される。しかし、従来のマシンコード生成フローでは、実施の形態1におけるマイクロコントローラのようにマシンコードに周波数制御信号を付加することはできない。

#### 【0065】

本実施の形態3のマシンコード生成フローでは、プログラムソース内に分周比設定記述を挿入することでマシンコードに周波数制御信号を付加することができる。例えば、図7(b)のプログラム例のように命令①、③、④、⑥、⑦の順にプログラミングし、命令①、⑥、⑦は1分周で実行し、命令③、④は2分周で実行させたい場合、命令①の前段に‘set fast’と記述し、命令③の前段に‘set slow’、命令⑥の前段に‘set fast’と記述する。コンパイラは命令とその前段

の 'set fast' あるいは 'set slow' の情報を入力として、命令をマシンコードに変換し、さらに前段の分周設定記述が 'set fast' であれば '0' をオペコードに付加し、'set slow' であれば '1' を付加する。また、命令の前段に何も記述がなければ前のマシンコードの付加信号を継続する。

#### 【0066】

すなわち本実施の形態3のマイクロコントローラは、前述した実施の形態1の構成において、上述のコンパイラと、コンパイラで生成したマシンコードをROMコードに変換する手段と、ROM配置する手段とを備えたものである。

#### 【0067】

以上のように本実施の形態3によれば、周波数制御信号のオペレーションコードへの付加は、プログラム開発者の負担なくコンパイラによって自動でマシンコード化が可能となり、実施の形態1のマイクロコントローラの特徴を活かすことができる。

#### 【0068】

(実施の形態4)

本発明の実施の形態4のマイクロコントローラについて説明する。

#### 【0069】

図8(a)は本実施の形態4のマイクロコントローラにおけるマシンコードの生成およびROMへの配置手順を示すフローチャート、図8(b)は本実施の形態4においてプログラムソースからコンパイラによってマシンコードを生成する方法を示す図、図8(c)は本実施の形態4のマイクロコントローラの動作タイミングチャートである。ROMへの配置手順に関しては実施の形態3と同様であるが、プログラムソースからコンパイラによってマシンコードを生成する過程において生成ルールが異なる。

#### 【0070】

実施の形態3では 'set fast'、'set slow' によってオペコードに付加する周波数制御信号を決定するが、実施の形態1のマイクロコントローラでは分周比を変更する場合、オペコードに周波数制御信号を付加しているため、フェッチからデコード、実行し、osc sel が切り替わるまでにやや時間を必要とする。

そのため本来分周比を切り替えたい命令の実行タイミングに間に合わなくなる場合があり、プログラム開発者による周波数制御の詳細検討が必要となる。

#### 【0071】

本実施の形態のマシンコード生成フローでは、実施の形態3と同様に命令①から⑦までのマシンコードを生成した場合、分周比設定記述とその前段の命令の実行サイクルをもとに周波数制御信号を決定する。

#### 【0072】

例えば命令③の前段の‘set slow’は本来命令③を2分周で動作させる目的で挿入されているが、実施の形態1においては図3のようにオペコードに付加されているためoscse1の切り替わりが遅れ、実際には命令④の実行サイクル（ $④\mu - 1$ ）から切り替わることになる。この場合の対策として、命令③を2分周で実行するためにはその前の命令①に2分周制御の付加信号をもたせることによってちょうど $③\mu - 1$ の実行タイミングGで2分周に切り替わる。ただし、命令①の実行サイクルが長い場合には命令①の処理周波数が2分周となってしまう。本実施の形態4においては、‘set fast’、‘set slow’の前段の命令の実行サイクルが基準サイクル未満であれば前段の命令のマシンコードに付加し、基準サイクル以上であれば後段の命令のマシンコードに付加する。図8の例では、基準サイクルを3サイクルとし、‘set fast’、‘set slow’の前段の命令の実行サイクルが3サイクル未満（すなわち2サイクル以下）であれば前段の命令のマシンコードに付加し、3サイクル以上であれば後段の命令のマシンコードに付加するようにしている。

#### 【0073】

すなわち本実施の形態4のマイクロコントローラは、前述した実施の形態1の構成において、上述のコンパイラと、コンパイラで生成したマシンコードをROMコードに変換する手段と、ROM配置する手段とを備えたものである。

#### 【0074】

以上のように本実施の形態4によれば、周波数制御信号のオペレーションコードへの付加は、命令の実行サイクル数に応じて決定するため、より正確に意図するタイミングで周波数の制御が可能となり、プログラム開発者が実行サイクル数

を考慮してプログラミングする必要がないためプログラム開発効率の向上が可能となる。

#### 【0075】

##### (実施の形態5)

本発明の実施の形態5のマイクロコントローラについて説明する。

#### 【0076】

図9(a)は本実施の形態5のマイクロコントローラにおけるマシンコードの生成およびROMへの配置手順を示すフローチャート、図9(b)は本実施の形態5においてプログラムソースからコンパイラによってマシンコードを生成する方法を示す図である。ROMへの配置手順に関しては実施の形態3と同様であるが、プログラムソースからコンパイラによってマシンコードを生成する過程において生成ルールが異なる。

#### 【0077】

本実施の形態のマシンコード生成フローでは、プログラムソース内に分周比設定記述を挿入することで設定した分周比に対応する拡張コードを選択し、分周比設定記述の後段の命令はその拡張コードによって分類された命令マップの中から相当するマシンコードに変換される。例えば、図9(b)のプログラム例のように命令①、③、④、⑥、⑦の順にプログラミングし、命令①、⑥、⑦は1分周で実行し、命令③、④は2分周で実行させたい場合、命令①の前段に‘set fast’と記述し、命令③の前段に‘set slow’、命令⑥の前段に‘set fast’と記述する。命令マップは1分周で実行する命令群は拡張コードなし、拡張コード‘0010’、拡張コード‘0011’で、2分周で実行する命令群は拡張コード‘0100’、拡張コード‘0101’、拡張コード‘0110’で分類されるとする。本実施の形態5のマイクロコントローラで扱われる命令は実施の形態2で説明した図5の例のように1分周の命令群と2分周の命令群は拡張コード以外は同一のオペコードを持ち、同一の動作をするものとする。コンパイラは命令とその前段の‘set fast’あるいは‘set slow’の情報を入力として、前段が‘set fast’であれば1分周で実行する命令群のマップのマシンコードに変換し、‘set slow’であれば2分周で実行する命令群のマップのマシンコードに変換する。ま

た、命令の前段に何も記述がなければ分周比の変更はないものとし、前段の命令と同じ分周比の命令群のマシンコードに変換する。

#### 【0078】

すなわち本実施の形態5のマイクロコントローラは、前述した実施の形態2の構成において、上述のコンパイラと、コンパイラで生成したマシンコードをROMコードに変換する手段と、ROM配置する手段とを備えたものである。

#### 【0079】

以上のように本実施の形態5によれば、分周比の設定は分周比設定記述の挿入のみでマシンコードに反映することが可能となり、実施の形態2のマイクロコントローラの特徴を最大限活かすプログラム開発環境を提供できる。

#### 【0080】

##### 【発明の効果】

以上のように、本発明によれば、周波数を制御するための情報をオペレーションコードに付加あるいは拡張コードに保有させることで、実行命令のパイプライン処理の中で周波数制御が可能となるため、周波数変更に要するタイムラグを解消しタスクに応じた周波数制御を的確かつ迅速に実行することによりマイクロコントローラの消費電力を低減することができる。

##### 【図面の簡単な説明】

##### 【図1】

本発明の実施の形態1のマイクロコントローラの概略構成を示すブロック図

##### 【図2】

本発明の実施の形態1のマイクロコントローラの概略構成を示すブロック図

##### 【図3】

本発明の実施の形態1のマイクロコントローラの命令フォーマット、プログラム例および動作タイミングチャート

##### 【図4】

本発明の実施の形態2のマイクロコントローラの概略構成を示すブロック図

##### 【図5】

本発明の実施の形態2のマイクロコントローラの命令マップ説明図



**【図 6】**

本発明の実施の形態 2 のマイクロコントローラの命令フォーマット、プログラム例および動作タイミングチャート

**【図 7】**

本発明の実施の形態 3 のマイクロコントローラにおける ROM コード生成処理を示すフローチャートとその説明図

**【図 8】**

本発明の実施の形態 4 のマイクロコントローラにおける ROM コード生成処理を示すフローチャートとその説明図および動作タイミングチャート

**【図 9】**

本発明の実施の形態 5 のマイクロコントローラにおける ROM コード生成処理を示すフローチャートとその説明図

**【図 10】**

従来のマイクロコントローラの一般的な概略構成を示すブロック図

**【図 11】**

従来のマイクロコントローラの一般的な概略構成を示すブロック図

**【図 12】**

従来のマイクロコントローラの命令フォーマット、プログラム例および動作タイミングチャート

**【符号の説明】**

100 CPU

200 バスインターフェイス

300 データパス

400 命令デコーダ

401 インストラクションフェッチバッファ (IFB)

402 インストラクションバッファ (IB)

403 インストラクションレジスタ (IR)

404 プログラマブルロジックアレイ (PLA)

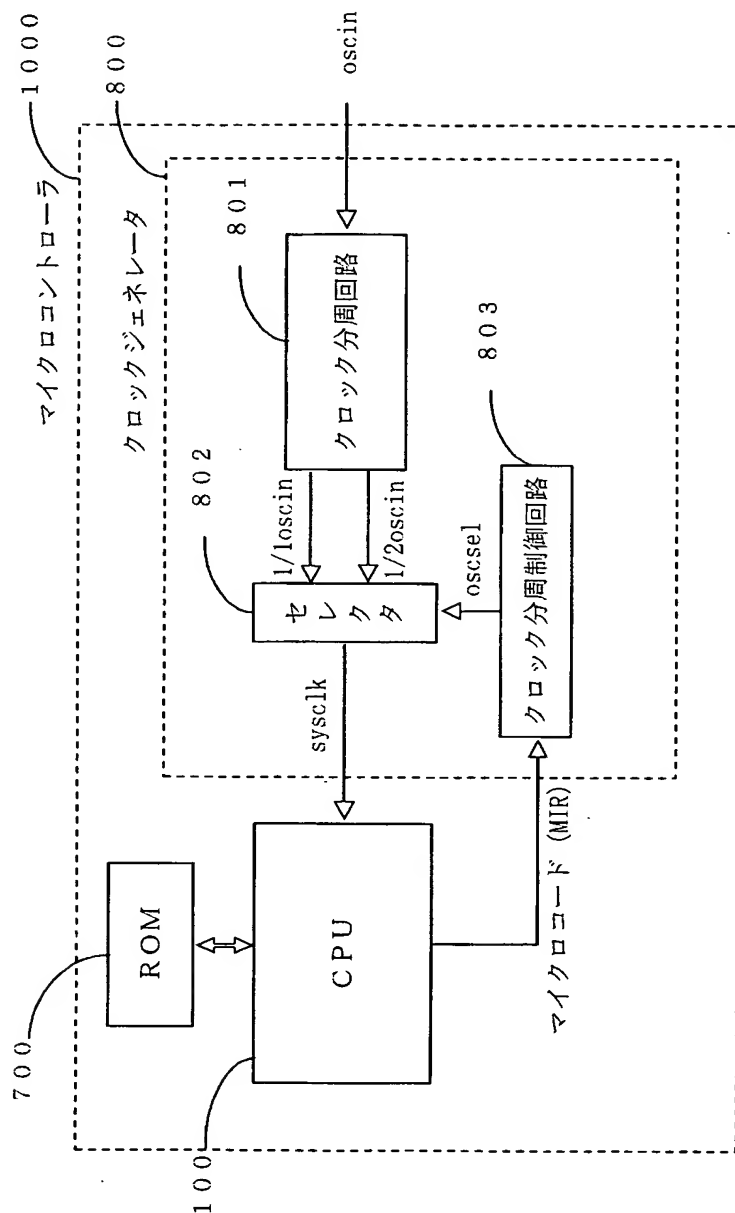
500 データレジスタ

600 アドレスレジスタ  
700 ROM  
800 クロックジェネレータ  
801 クロック分周回路  
802 セレクタ  
803 クロック分周制御回路  
804 クロック分周制御レジスタ  
1000 マイクロコントローラ

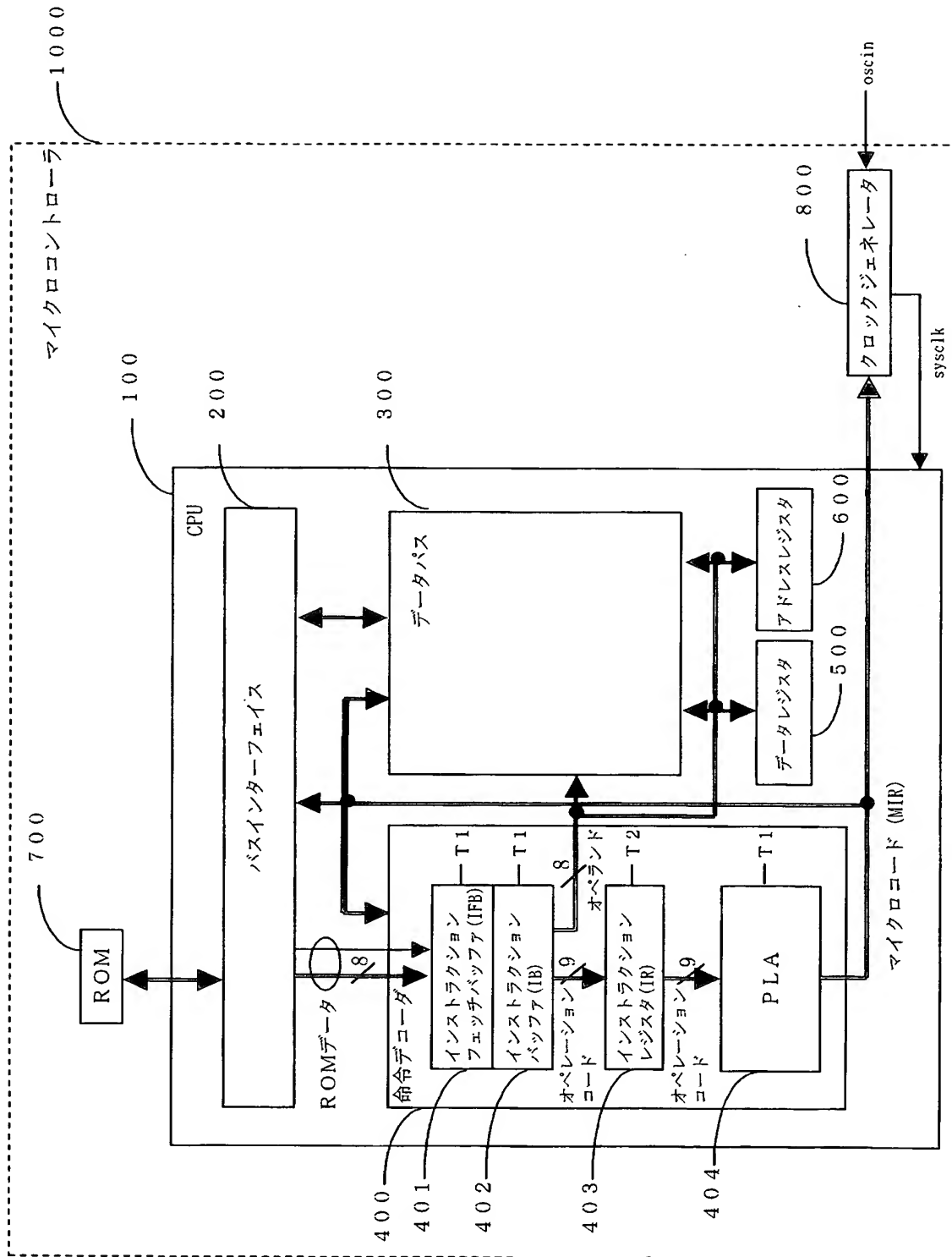
【書類名】

図面

【図 1】

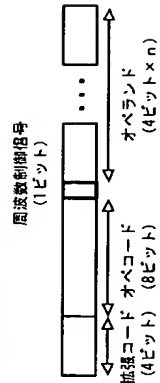


【図 2】



【図 3】

(a) 命令フォーマット

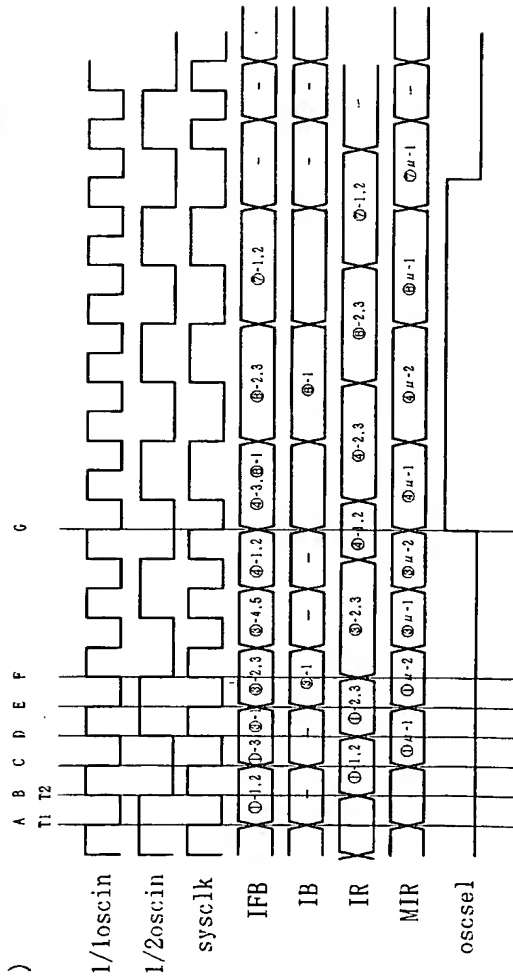


(b) 命令 マシンコード (ニブル単位)

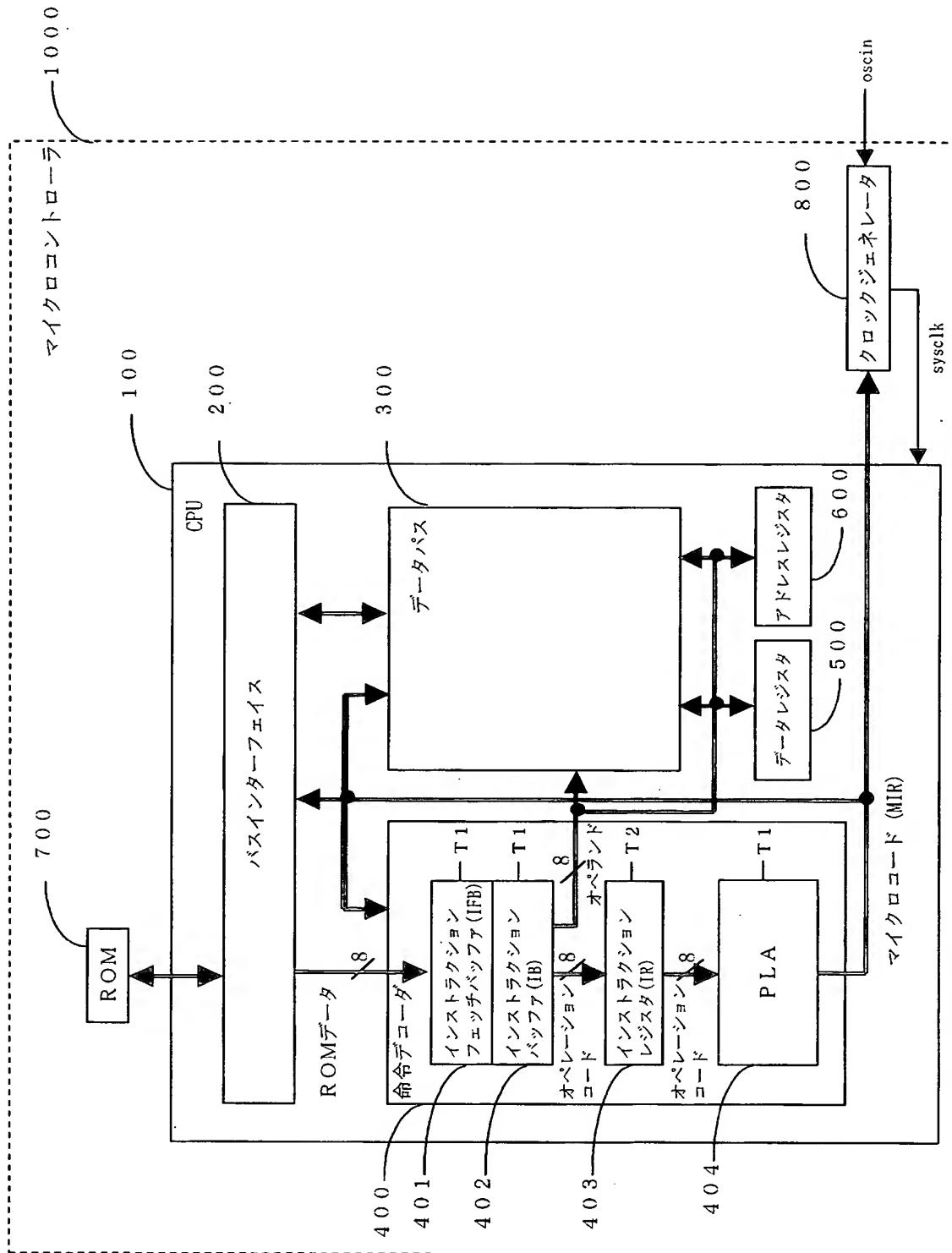
命令	1	2	3	4	5
①ADD D0, D1	(0011) 0111	0001*0	3ニブル2サイクル命令	*D0, D1: データレジスタ	(0011) 拡張コード
③SUB imm8, D1	(0010) 1010	0101*1 <#8...>	5ニブル2サイクル命令		(0010) 拡張コード
④SUB D0, D3	(0010) 1010	0011*1	3ニブル2サイクル命令	*D3: データレジスタ	(0010) 拡張コード
⑥ADD D2, D3	(0011) 0011	0001*0	3ニブル1サイクル命令	*D2: データレジスタ	(0011) 拡張コード
⑦AND D2, D3	(0011) 0111	1011*0	3ニブル2サイクル命令		(0011) 拡張コード

\*0: 分周制御信号、sysclkをoscinの1分周に設定  
\*1: 分周制御信号、sysclkをoscinの2分周に設定

(c)



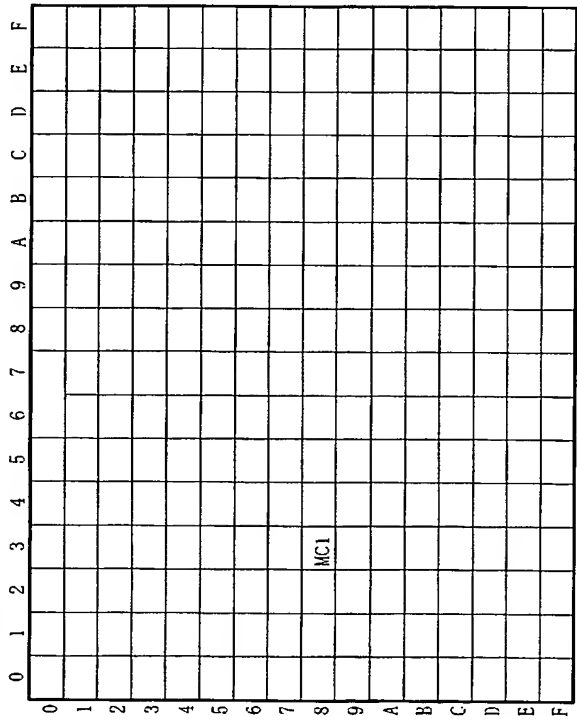
【図 4】



【図 5】

(a) 命令マップ A (1 分周対応)

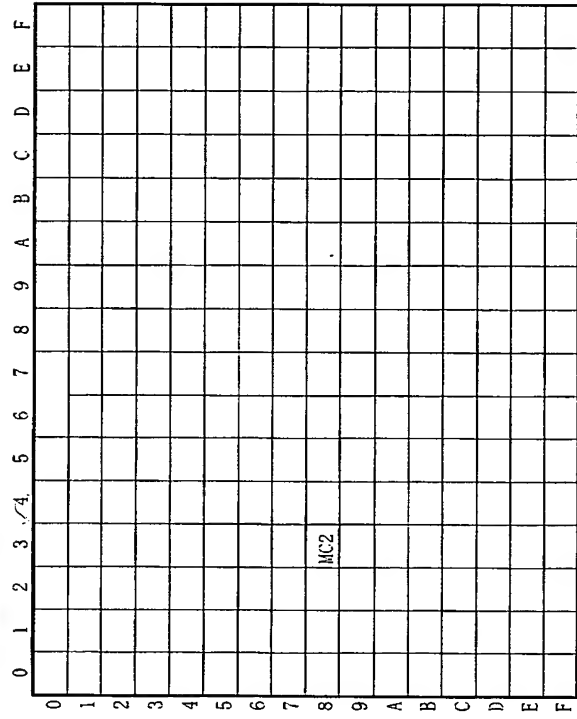
拡張コード'b'0011'  
第 1 ニブル/第 2 ニブル



MC1: 0011 1000 0011 .....  
1 分周    オペコード    オペランド  
拡張コード

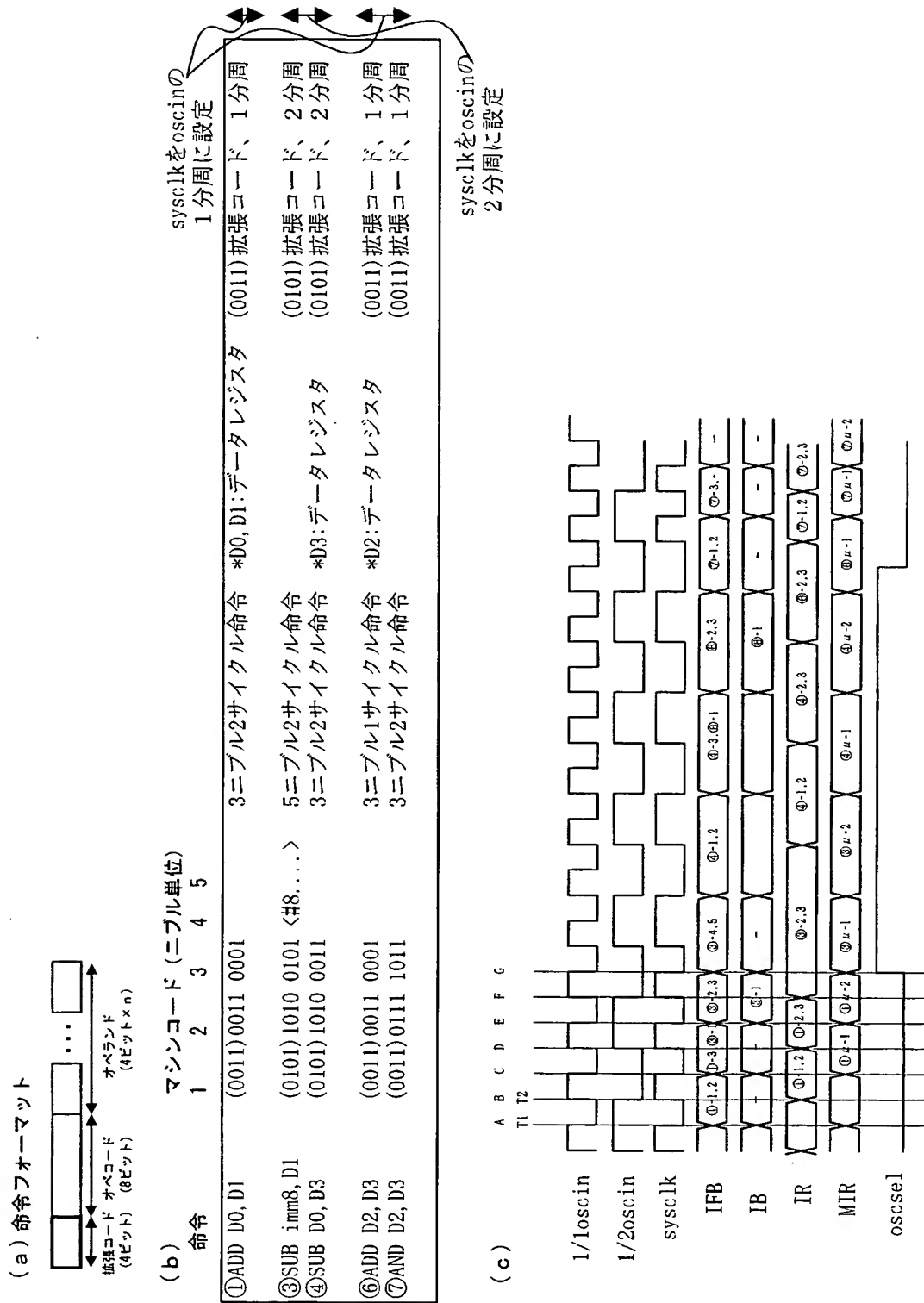
(b) 命令マップ B (2 分周対応)

拡張コード'b'0101'  
第 1 ニブル/第 2 ニブル



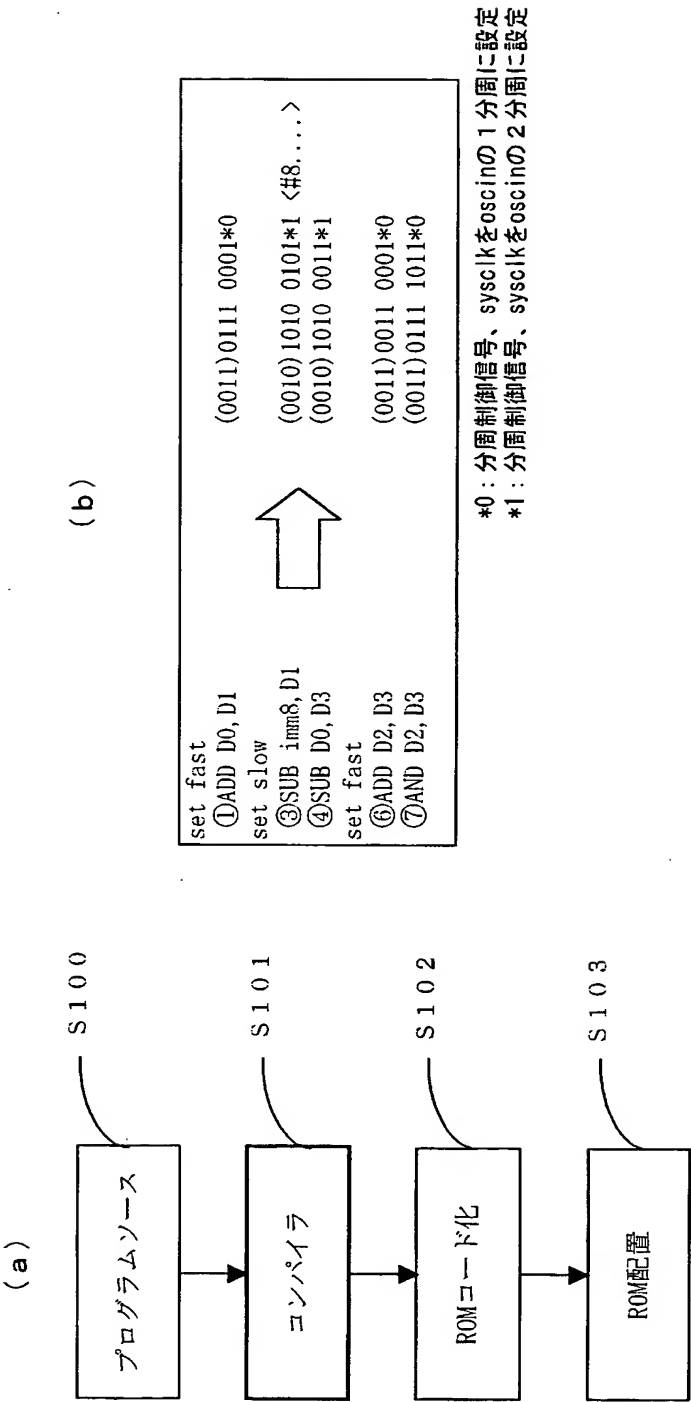
MC2: 0101 1000 0011 .....  
2 分周    オペコード    オペランド  
拡張コード

【図 6】

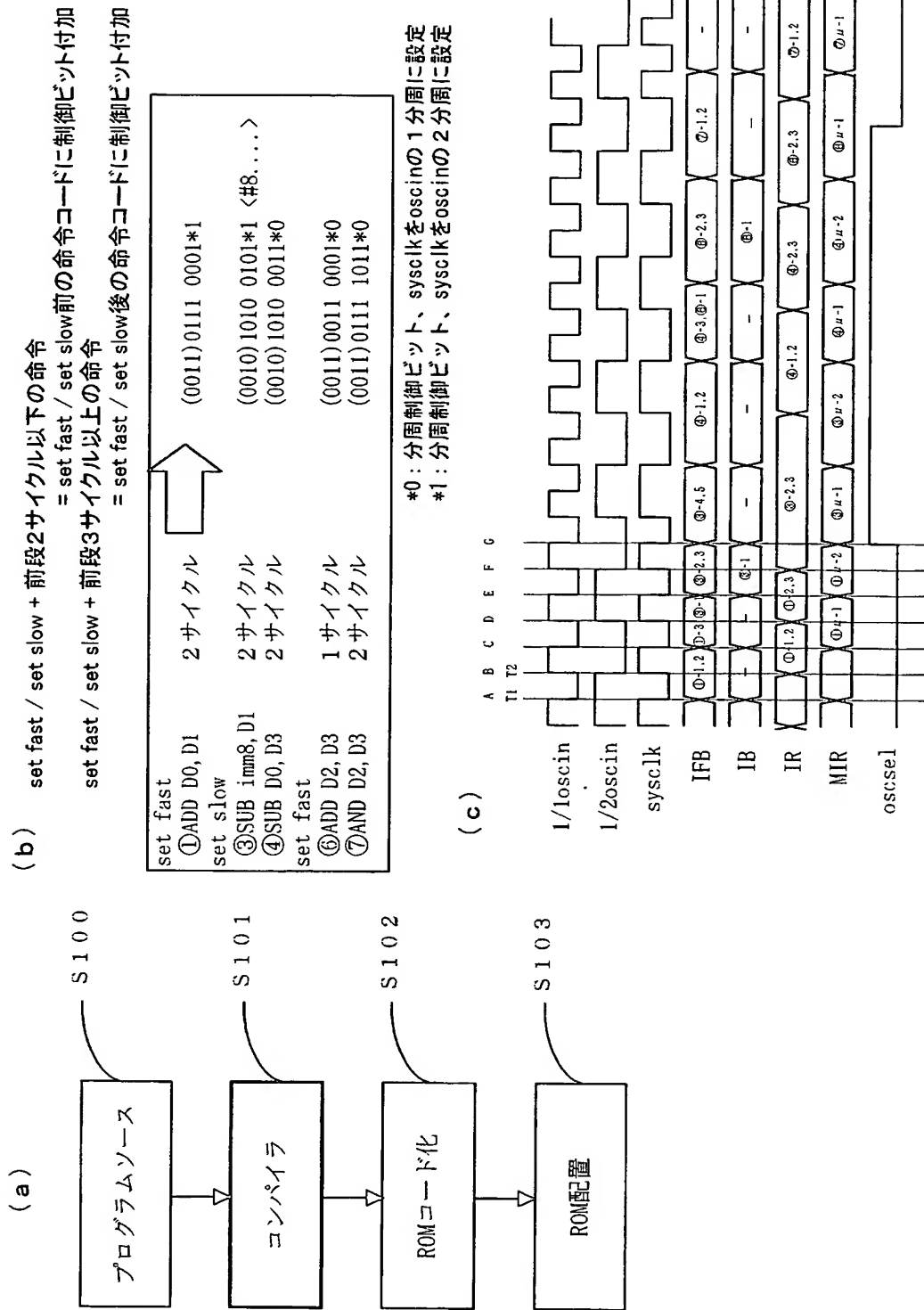




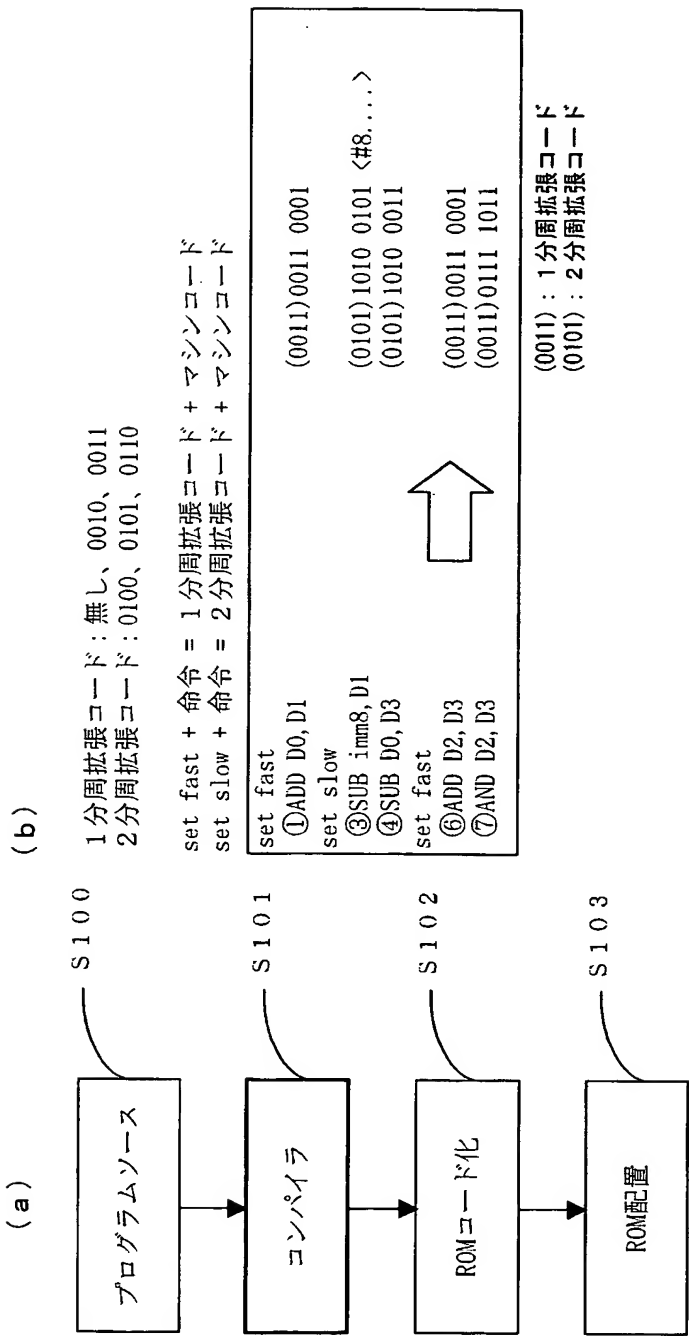
【図 7】



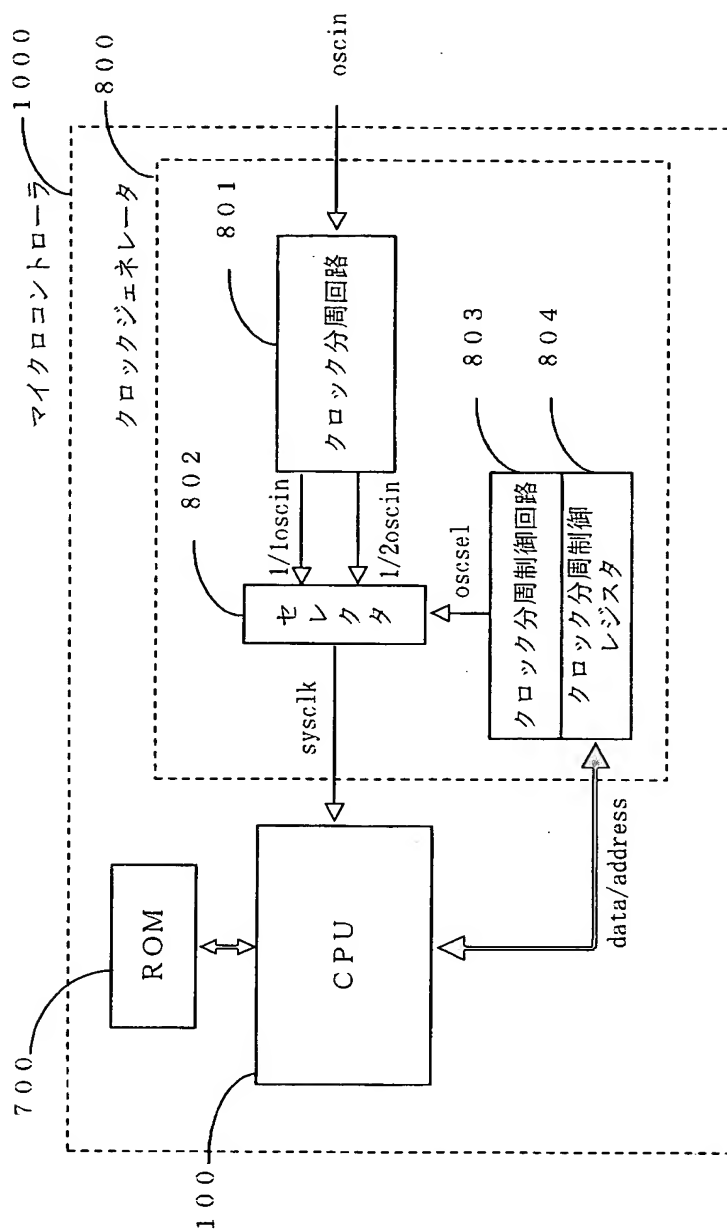
【図 8】



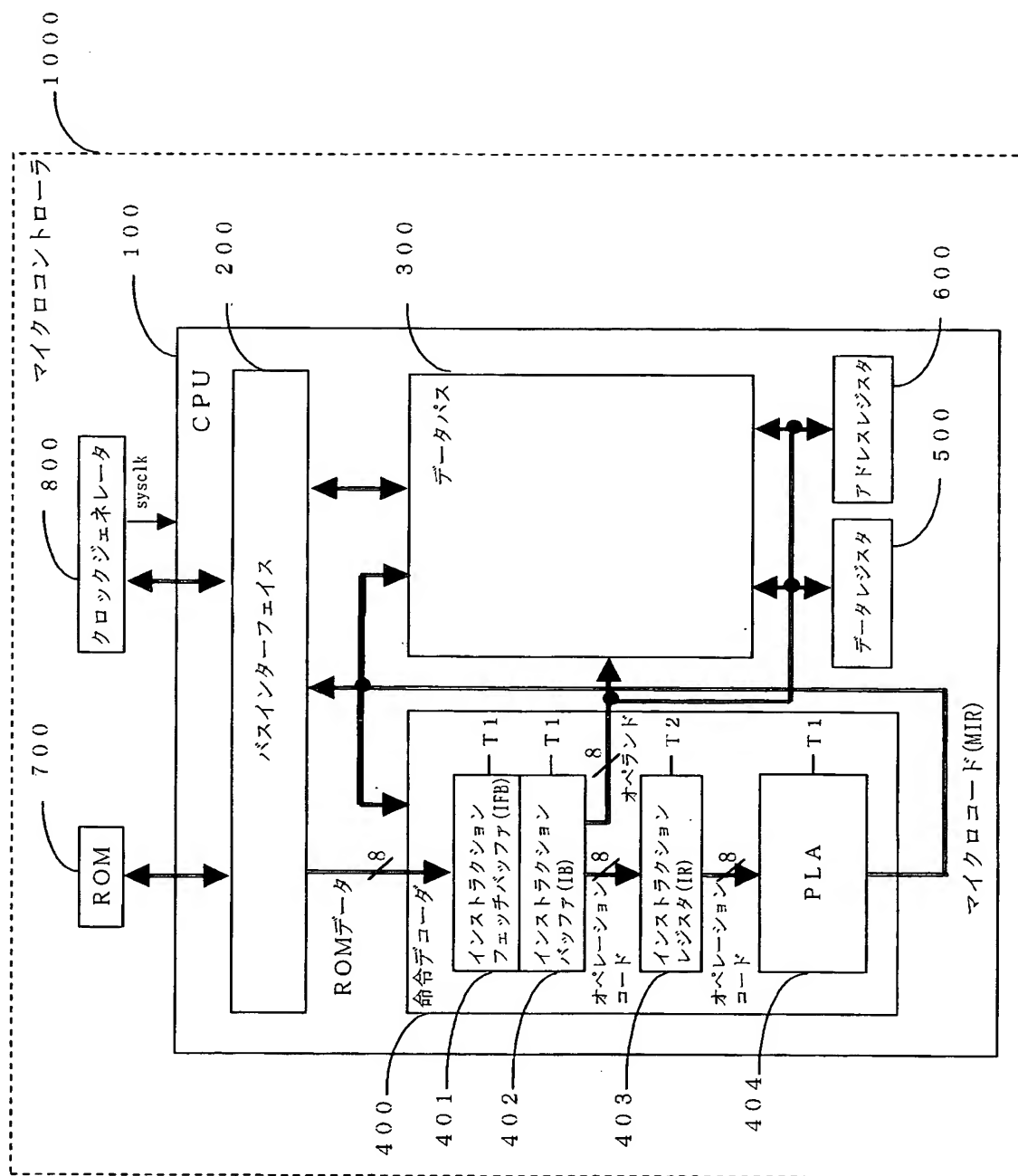
【図 9】



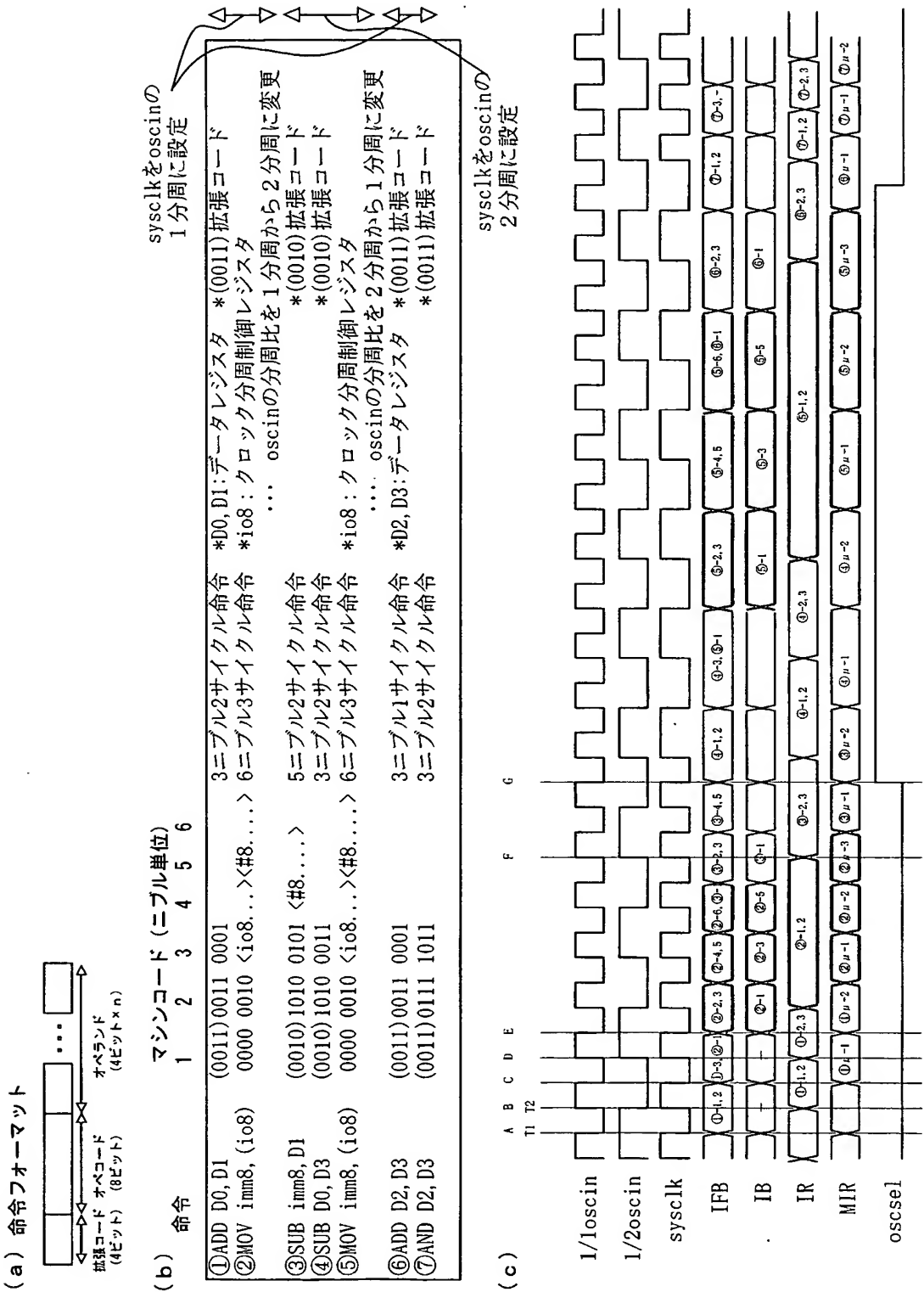
【図 10】



【図11】



【図 12】



【書類名】 要約書

【要約】

【課題】 動作中の周波数変更に要するタイムラグを解消することができ、タスクに応じた周波数制御を的確かつ迅速に実行することにより、消費電力を低減化することができるマイクロコントローラを提供する。

【解決手段】 オペレーションコードにシステムクロックの分周比を決定する周波数制御信号を付加し、その周波数制御信号をオペレーションコードとともにパイプライン処理によりフェッチ、解読することにより、実行命令のパイプライン処理の中で周波数制御が可能となるため、周波数変更に要するタイムラグを解消しタスクに応じた周波数制御を的確かつ迅速に実行することによりマイクロコントローラの消費電力を低減することが可能となる。

【選択図】 図3

認定・付加情報

特許出願の番号	特願 2 0 0 3 - 0 9 4 6 8 6
受付番号	5 0 3 0 0 5 3 0 1 3 5
書類名	特許願
担当官	第七担当上席 0 0 9 6
作成日	平成 1 5 年 4 月 1 日

< 認定情報・付加情報 >

【提出日】	平成15年 3月31日
-------	-------------

次頁無



特願 2 0 0 3 - 0 9 4 6 8 6

出 願 人 履 歴 情 報

識別番号

[ 0 0 0 0 0 5 8 2 1 ]

1. 変更年月日

1 9 9 0 年 8 月 2 8 日

[変更理由]

新規登録

住 所

大阪府門真市大字門真 1 0 0 6 番地

氏 名

松下電器産業株式会社